# VisualHaptics 2.0: 感触の解像度向上とタッチパネルへの応用

小島諒介 $^{\dagger 1}$  渡邊恵太 $^{\dagger 1}$  稲見昌彦 $^{\dagger 1\dagger 2}$  五十嵐健夫 $^{\dagger 1\dagger 3}$ 

我々はカーソルの遅延や変形を用いて対象に触れているかのような感触を提示する VisualHaptics を開発してきた. 本研究では感触の解像度を高める手法 VisualHaptics HD とタッチパネル環境で感触提示を実現する VisualHaptics Touch から成る VisualHaptics 2.0 提案する. VisualHaptics では対象の状態を 1 点(1 ピクセル)にて認識し、それに基づき カーソルの形状が変形する. 本研究ではカーソルが対象を面で捉えるようにすることで、対象の状態をカーソルから ユーザに伝達する分解能を高める. また、近年のユーザインタフェースはタッチパネル化が進んでおり、画面を指で直接指示するため、カーソルは画面上に表示されない. この際に VisualHaptics の手法は一見使えないと思われがちだが、VisualHaptics 2.0 では指と連動して動く画面側をカーソルとして捉え画面全体の動きをずらしたり遅延させたりすることによって、タッチパネル環境においても感触の提示を実現した.

# VisualHaptics 2.0: Improvement of the Feel and Applying to the Touch Screen

RYOSUKE KOJIMA<sup>†1</sup> KEITA WATANABE<sup>†1</sup> MASAHIKO INAMI<sup>†1†2</sup> TAKEO IGARASHI<sup>†1†3</sup>

We had been developing VisualHaptics, which presents the feel of touching to the target by modifying cursor icon and its motion. In this paper, we propose VisualHaptics 2.0. This is composed VisualHaptics HD that enhancing a resolution of feel and VisualHaptics Touch that enables present feel on the touch screen without cursor. Previous VisualHaptics had transformed cursor based on only one reference point to detect context of target. VisualHaptics HD transforms cursor based on multiple points to detect the target. In addition, with the increased usage of touch screen in recent years, it had not been used cursor in GUI. At first sight, we think we could not use the VisualHaptics method, however we regard whole screen as cursor, because moving of the screen linked users finger. Therefore we can also present the feel on touch screen without cursor.

# 1. はじめに

筆者らはカーソルの遅延や変形を用いて対象に触れているかのような感触を提示する VisualHaptics を開発してきた [1,2,3]. またこのような視覚情報を用いて、感覚情報を提示する手法として Pseudo-Haptics[4]がある. VisualHaptics は、テクスチャのざらざら感や、抵抗感、べとつきといった感覚を、マウスから入力される位置とカーソルの表示位置をずらすことで実現する. また画面に表示される対象の面に合わせてカーソルを変形することで、対象をなぞり触れているかのような感覚の提示を実現する.

我々はこれらの発展として VisualHaptics の改良: VisualHaptics HD とタッチパネルへの応用: VisualHaptics Touch を試作し, さらにタッチパネル環境でユーザが

# 1.1.1 VisualHaptics の高解像度化

我々は VisualHaptics を通じて、カーソルに対して変形や 遅延などの効果を与えると、操作者の体験がさまざまに変 化すると考察した. 特にカーソル設計方法によって体験が 変化に重要であると考察した. そこで、本研究では「カー ソルの変形の自由度」の調整によって、感触の解像度をコ ントロールする VisualHaptics HD を提案する.

従来の VisualHaptics では、カーソルの先端部分がある対象の 1 点の状態を取得し変形する. したがって、たとえば立方体の角の部分にカーソルを乗せると、図 1 のように対象の状態に沿わないように見えてしまう. これは、カーソルが 1 点で対象(環境)を参照し、かつカーソルのグラフィックスのモデルデータが単純であることが問題である. この状態での理想は、この状態では、カーソルが折れ曲がっているように見えることである. そこで、VisualHaptics 2.0 ではカーソルが多点で対象を参照し、その参照に基づきカーソルが各参照部分に応じて変形する VisualHaptics HD を実現する(図 2).

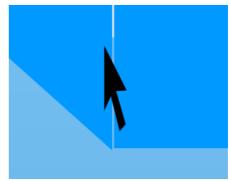


図 1 従来の Visual Haptics では角の部分で変形しない. カーソルの見た目が折り曲がるように変形する方が理想

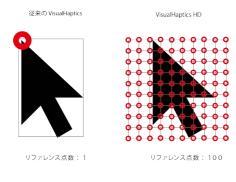
<sup>†1</sup>JST ERATO 五十嵐デザインインタフェースプロジェクト JST ERATO IGARASHI UI Porject

<sup>†2</sup> 慶應義塾大学

Keio University

<sup>†3</sup> 東京大学

University of Tokyo



○ リファレンス点

図 2 従来の VisualHaptics と VisualHaptics HD. リファレンス点数が異なる.

# 1.1.2 VisualHaptics のタッチパネルへの応用

近年デスクトップやラップトップコンピュータから, タッチパネルを搭載したタブレット型のコンピュータが広く普及しはじめた.

タッチディスプレイを用いたインターフェースでは必ずしもカーソルがあるとは限らない.カーソルがない場合において VisualHaptics の手法は一見使えないと思われがちだが、指と連動して動く画面側をカーソルとして捉え画面全体の動きを変化させることで同様の効果が期待できる.またそれとは別に、タッチインターフェースにおいてもアイコンのような仮想の物体が指と連動して動くという場面では、同様にカーソルとマウスの関係がアイコンと指の間にも成り立つのではないかということが考えられる.

加えて、タッチインターフェースで広く使われている操作方法にピンチ動作による拡大・縮小と指をスクリーンから離した後にも直前の動作が継続する「慣性」がある.これらの動作についても VisualHaptics の拡張を考える.ここではこれらのタッチパネルへと拡張した VisualHaptics を VisualHaptics Touch と呼ぶ.

#### 2. VisualHaptics 2.0

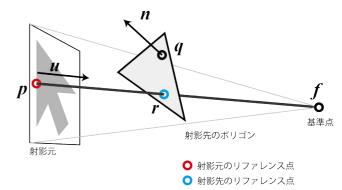
VisualHaptics 2.0 は VisualHaptics HD と VisualHaptics Touch からなり、ここではそれらの具体的な実現方法を説明する.

# 2.1.1 VisualHaptics HD の基本部分の実装

VisualHaptics HD を実装するに当たり、我々は感覚情報を効果的に提示するためには次の2点が重要であると考える。一つは感覚提示の強度を調節できるという点である。これにより、カーソルが指し示す対象やコンテンツごとに強度の異なる感覚を提示することが可能になり、より表現の幅が広がると考えられる。もう一つは、ユーザが不自然に感じない程度のなめらかな動作であるという点である。VisualHaptics HD では三次元空間上での操作が入るため、実装によっては処理に時間がかかる場合もある。しかし、VisualHaptics はユーザとのインタラクションにより実現されるため、リアルタイムに処理できることが重要である。

そのため, 処理速度に注意しながら実装を行う必要がある.

これらの点を踏まえ、我々は VisualHaptics HD の実装として対象への射影を用いた手法を提案する。図 3 は射影の方法を図示したもので、具体的なアルゴリズムについて以下で説明する。まず、三次元空間に配置したカーソルの画像を格子上に分割する。次にそのそれぞれの格子点を対象に射影する。ここでは、対象は三角形ポリゴンの集合で定義されていると仮定する。この時、格子点の三次元空間位置ベクトルをp,任意の基準点の位置ベクトルをfとして、線分f,pと交差している対象の三角形ポリゴンを射影先の面として選択する。



p:射影元のリファレンス点の座標 r:射

 $m{r}$ :射影先のリファレンス点の座標  $m{q}$ :射影先のポリゴン上の点の座標

**u**:射影する向き **f**:基準点の座標

n:射影先のポリゴンの法線

図 3VisualHaptics HD の射影方法

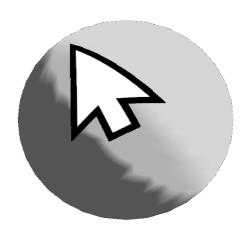


図 4 VisualHaptics HD の球面への適用例 カーソルが球面に張り付いたような見た目になる.

線分f,pと交差するポリゴンが複数ある場合には線分f,pとの交点が最も fに近い面を射影先の面として選択する. 最終的に求めるべき(点pの)射影先rは線分f,pと射影 先の面との交点として定義する.これにより射影先の面を 三次元座標qと単位法線ベクトルnで表すとすると,三次 元座標位置ベクトルrは以下のような式で表現できる.

$$r = \frac{\mid (p - q) \cdot n \mid}{\mid n \cdot u \mid} u + p$$

ただし、 $\mathbf{u}$  は次の式で計算される単位ベクトルである.

$$u = \frac{(f - p)}{|f - p|}$$

すべての格子点においてこの式により射影先の点を求めることができる.また,予め格子点の画像上の座標(uv 座標)を保存しておき,射影先の点にも同じ座標を割り当てる事によって画像を射影先に再構成できる.この方法を用いて球面上にカーソル画像を射影したものが図 4 である.

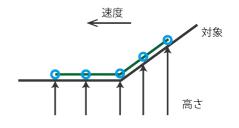
この実装法の特徴として奥行きと解像度の調節が可能という点がある。まず、奥行きは VisualHaptics において対象に触れているかのような感覚を表現するための重要な要因である。この奥行きを調節するためのパラメータが基準点の位置ベクトル f である。例えば、基準点と射影元の面の距離を小さくとると、物体の遠近感をより強調することができる。次に、解像度は格子点の数に対応するため、格子点の数を変えることで解像度を調節することができる。また、解像度の調節によって処理時間とのトレードオフを図ることができるような実装になっている。

この実装法の問題として二つの点が挙げられる. ひとつは対象の角においてカーソルが不自然に見える場合があるという点である. 格子点が有限であるために, 対象の角をうまくとらえることができずに, カーソルが対象に埋まってしまったり角が丸身を帯びて表現されてしまったりする. これは格子点を十分に細かく取ることによって防ぐことができる. カーソルが対象に埋まる問題については別の方法として, 実際の対象とは別に射影するために一回り大きなモデルを用意するなどの方法で緩和できる. もう一つの問題はカーソルが引き伸ばされて不自然に見える場合があるという点である. この問題は射影を用いている事による問題であり, 斜めから差し込む光によって影が伸びるといった自然現象でもよく見られるが, カーソルにおいては不自然に見える場合がある. 特に急峻なモデルや凹物体においてよく起こる現象である.

#### 2.1.2 VisualHaptics HD における感触表現の実装

VisualHaptics HD において感触表現をどのように実装するかについて説明する.「ざらざら感」については従来のVisualHaptics と同様にカーソルにノイズを加えることによって表現している.また、従来のVisualHaptics のようなカーソルの遅延や移動速度を用いる方法についてはVisualHaptics HD でも仮想的なカーソルを用いることができるため簡単に実装できる.従来のVisualHapticsではなかった表現として、VisualHaptics2.0では「滑り」の表現を行った.「滑り」については三次元の特性を生かした方法で表現することができる.三次元のモデルの起伏を読み取り、それに合わせてカーソルに速度を加える事により滑りやすいという表現を行う.実装としては、図5のように、格子点を射影した先の点すべてについて「高さ」を計算する.隣接する点との「高さ」の差分を格子の縦と横の両方につ

いて計算し、それぞれの方向について差分を平均する.



- 射影先のリファレンス点
- ─ 射影したカーソル

図 5「滑り」の表現方法

「高さ」に応じた速度をマウスからの入力速度に加える.

これは傾きの平均を近似的に計算していることに相当する. この量に比例する速度をマウスの速度に加えてカーソルを 動かすという方法で「滑り」を実装している.

#### 2.1.3 VisualHaptics Touch の実装

VisualHaptics Touch の感覚表現には、アイコン等の仮想的な物体を通しての表現と画面全体での表現がある。そこで、アイコンと画面全体の両方について「重さ」・「軽さ」・「ざらざら感」の実装を行った。また、動作についてもアイコンや画面全体をスライドさせる動作、ピンチ動作による拡大・縮小、指をスクリーンから離した後の「慣性」といったものがあるがこれらの動作に応じて実装を行った。

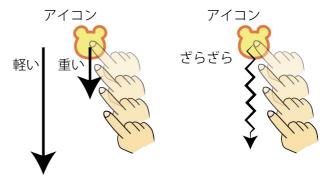


図 6アイコンを用いた感覚の表現 実際の指の動きとアイコンの動きの違いにより表現する.



図 7 画面全体を用いた感覚の表現 実際の指の動きと画面の動きの違いにより表現できる

まず、スライドさせる動作における「重さ」・「軽さ」の

実装は指の動きと比較しての相対的な速度によって表現する。例えばアイコンでは、アイコンを指でスライドさせて動かすという場合、指の動きとアイコンの動きが完全に一致するというのが自然である。しかし、この時に指の動きに比べてより早い速度で動いた場合にはアイコンが「軽く」なったような印象を受け、逆に指の動きに比べて遅い速度で動いた場合にはアイコン「重く」なったような印象を受けるのではないかと考え、そのように表現した(図 6 左)。画面全体の場合もこれと同様に、指の動きとスクロールの動きの相対的な速度で「重さ」・「軽さ」を表現した(図 7 左)。

次に、ピンチ動作による拡大・縮小における「重さ」・「軽 さ」の実装も拡大率・縮小率を変化させることで表現した.

また,「慣性」については指が離れた後もしばらくうごきが継続するというものであるが,この継続時間を短くすることで「重さ」を,長くすることで「軽さ」を表現した.

最後に、「ざらざら感」については、スライド動作、ピンチ動作、「慣性」、のすべての動作について従来のVisualHaptics と同様にノイズを加えることによって実現した。ただし、アイコンを通しての「ざらざら感」についてはアイコン自体がざらざらしているような場合とアイコンを通してその裏側の背景部分がざらざらしている場合の二通りが考えられるが、今回はその両方について実装した。(図 6 右、図 7 右)

# 3. 考察と応用

#### 3.1.1 感触と解像度

解像度をあげることにより触っている感覚がさらに増すのではないかと予想したが、実際には解像度をあげることで別の表現を可能にしたという側面が強かったように感じられた。これは、従来の VisualHaptics では一点で対象を参照しているのに対し VisualHaptics HD では面に近い形で参照しているため、従来の単純に触るという感覚ではなく、「撫でる」という感覚に近い。そして、カーソルの解像度を低くすると、カーソルの折れ面がわかりやすくなる。筆者らの体験では、このときの感覚を「ぱきぱき感」と呼んだ。また、今回はカーソルの形状は一点を指す矢印型で実装を行ったが、VisualHaptics HD でより効果的に表現するためのカーソルの形状については今後検討が必要である。

#### 3.1.2 タッチインターフェースと VisualHaptics

VisualHaptics Touch の実装を通して、VisualHaptics はタッチインターフェースに適応が可能だということがわかった。また、タッチインターフェースで広く使われつつあるピンチ動作や「慣性」との連携についても行うことができた。今後、これらの具体的な評価について行っていきたい。

## 3.1.3 Web デザインへの応用

VisualHaptics2.0 によりタッチインターフェースにおいても VisualHaptics を用いることができるということを示し

た. VisualHaptics 2.0 も従来の VisualHaptics 同様にマウスやタッチといった標準的なインターフェースのみでユーザに感触情報を提示できるという特徴を持つ. これは、Webページにおける表現方法として容易で、有効である事を示唆している. そこで我々は現在 Web デザイナが感触デザインできるライブラリ VisualHaptics Lib を開発している. 近年、HTML5 や css3 により Webページの表現力が格段に大きくなった. VisualHaptics Lib はこれらの機能を用いて、感触デザインのための各種機能を Java Script として提供していく.

# 4. おわりに

本研究では Visual Haptics の発展としてカーソルの変形の自由度を高めることによる感触の分解能の向上を目的にした Visual Haptics HD と,画面全体をカーソルと見なすことによる Visual Haptics のタッチパネルへ応用した Visual Haptics Touch の提案(Visual Haptics 2.0)の試作を行い、その実装について述べた. さらに、Web デザイナが簡単に感触のデザインを行えるようにするため、Javascript による Visual Haptics Lib の取り組みについて紹介した.

# 参考文献

- 1) Keita Watanabe and Michiaki Yasumura. VisualHaptics: generating haptic sensation using only visual cues. In Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology (ACE '08). Pp.405-405. 2008.
- 2) 渡邊惠太, 安村通晃 RUI: Realizable User Interface カーソル を用いた情報リアライゼーション. 第 27 回ヒューマンインタフェース学会研究会「VR の心理と生理」. ヒューマンインタフェース 学会研究報告集, pp.35-38, 2004.
- 3) VisualHaptics Demo site http://www.persistent.org/VisualHapticsWeb.html
- 4) Lecuyer, A., Burkhardt, J., and Etienne, L: Feeling bumps and holes without a haptic interface: the perception of pseudo-haptic textures, CHI '04. ACM Press, pp.239-246. 2004.